Introduction To Assembly Language

شرح للعمليات الحسابية والمنطقية في لغة الاسمبلى

إعـداد : محمد عبد الفائدى



نبذة عن المؤلف

الاسم : محمد عبد ناجي الفائدى .

العنوان : ليبيا - بنغازي - الابيار

الدرجة العلمية :

بكالوريوس علوم الحاسوب - كلية تقنية المعلومات - جامعة قاريونس

بريد إلكتروني : Alfayde2002@yahoo.com

مهارات تقنية :

Programming Languages:

- Visual basic.net

Web Technologies:

- HTML

Database Management Systems

- Microsoft SQL Server 2000
- Microsoft Access

Conceptual knowledge:

- ERD ,DFD
- UML

General:

- Microsoft Office
- Adobe Photoshop

الإهداء

إلى إمام الذاكرين وقدوة السالكين ومعلم المتعلمين صاحب الشفاعة يوم الدين.

سيدنا محمد صلى الله عليه وسلم

إلى النبع الذي لا يميل العطاء .. يامن كان دعائها نوراً لي في طريق الوفاء .. وحبها ملأ الأرض والسماء.........أمي الحبيبة

إلي الظل الذي أظل حياتي بنصحه وإرشاده.. إلي من وهبني عمره لينير لي الطريق..إلي من هو لي القدوة والمثل..........أبي العزيز

إلي من دمي من دمهم وروحي من روحهم إلي من حبهم يجري في عروقي ... إلي نجوم حياتي المتلألأة دائما .... إخوتي الأعزاء

إلي الذين سعدت برفقتهم .. إلى من سطرت معهم على جدار الزمن أجمل الذكريات..إلي من عرفوا الصداقة بكل معانيها................ أصدقاء الدراسة .

إلى كل شخص بذل من جهده ووقته وفكره لكي ينشر معلومات أو مواضيع مفيدة على الشبكة العنكبوتية لكي يستفيد منها الآخرين.

مقدمة

بسم الله الرحمن الرحيم والصلاة والسلام على نبينا محمد وعلى اله وصحبه أجمعين.

أن اصدق الحديث كلام الله وخير الهدى هدى محمد صلى الله علية وسلام

أما بعد:

هذه الصفحات عبارة عن مجموعة لسلسة من المحاضرات أثناء الفترة الدراسية ومجموعة من الدروس على الانترنت قمت بتجميعها وشرحها ببساطه حتى يتمكن الباحث من فهمها في سهولة ويسر إن شاء الله.

وأردت أن يكون امتداداً لي بعد مماتي كما قال علية الصلاة والسلام :

(إذا مات ابن آدم انقطع عمله إلا من ثلاث: صدقة جارية، أو علم ينتفع به ، أو ولد صالح يدعوا له).

ولذلك قد أكون ميتاً ألان وأنت تقرأ هذه الصفحات، ولكني حياً الآن معك، أتحدث أليك وتستمع لي.

في هذه الصفحات لن تجد شفرات برمجية للغة الاسمبلى لأن الشرح يهتم من الناحية النظرية .

أسأل الله أن تكون هذا الصفحات مفيدة فما كان فيه من صواب فمن الله وما كان من

خطأ فمن نفسي والشيطان، والله الهادي إلى سواء السبيل.

العمليات الحسابية والمنطقية في لغة الاسمبلى

في لغة الاسمبلي هنالك عمليات حسابية مثل (الجمع ، الطرح ، الضرب ، القسمة) و هنالك العمليات المنطقية التي تتعامل مع البتات مثل

(OR , AND, NOT ,XOR ,TEST)

أولاً العمليات الحسابية

1) أمر الجمع ADD

الصيغة العامة لهذا ألامر هى ADD Destination,Source

ADD OP1, OP2

Op1=op1+op2

نفرض إن قيمة :-

OP1 تمثل الهدف (Destination)

OP2 تمثل المصدر (Source)

ملاحظات مهمه:

1: فى هذا الامر يتم جمع قيمة المصدر(OP2) وقيمة الهدف (OP1)وتحفظ النتيجة في الهدف(OP1).

2: بالنسبة للمصدر (OP2) والهدف(OP1) يجب ان يكون كلاهما من نفس الحجم (8 bit – 16bit).

3:بالنسبة للمصدر(OP2) يمكن ان يكون قيمة فورية أو سجل أو موقع فى الذاكرة .

4: بالنسبة للهدف(OP1) يمكن ان يكون سجل أو موقع فى الذاكرة.

5: بالنسبة للمصدر(OP2) والهدف (OP1)لايمكن ان يكون كلاهما موقع بالذاكرة.

مثال:

```
MOV DH,41H
ADD DH,3
```

الحل :

DH=DH+41

DH=3+41

DH=44H

بعد تنفيذ المقطع المسجل DH يصبح 44H

مثال :

MOV AL , 6

MOV CL , 8

ADD AL , CL

الحل:

AL= AL+ CL

AL= 6+8

AL=0EH

جدول متابعة:

AL	CL	الامر
؟؟	؟؟	
06H	؟؟	MOV AL , 6
06H	08H	MOV CL , 8
0EH	08H	ADD AL , CL

ملاحظات:

1: (؟؟) تعني اننا لا نعلم ما بداخله.

2: علينا تسجيل القيم في الميزان السادس عشر على شكل منزلتين في المسجلات ذات 8 بت وعلى شكل 4 منازل في المسجلات ذات 16 بت.

مثال:

AL=06H   DX=1024H

كل قيمة عددية تبدأ بحرف A , B ,C ,D ,E ,F يجب ان نسجل 0 عن يسارها

مثال:اكتب مقطع برنامج ينفذ ما يلي: CL=2*AL+BL

الحل:

```
MOV CL,AL
ADD CL ,CL
ADD CL,BL
```

2) أمر الطرح SUB

الصيغة العامة لهذا ألامر هى SUB Destination,Source

SUB op1 ,op2

op1= op1 – op2

نفرض إن قيمة :-

OP1 تمثل الهدف (Destination)

OP2 تمثل المصدر (Source)

ملاحظات مهمه:

1: فى هذا الامر يتم طرح قيمة المصدر(OP2) من قيمة الهدف (OP1)وتحفظ النتيجة في الهدف(OP1).

2: بالنسبة للمصدر (OP2) والهدف(OP1) يجب ان يكون كلاهما من نفس الحجم (8 bit – 16bit).

3:بالنسبة للمصدر(OP2) يمكن ان يكون قيمة فورية أو سجل أو موقع فى الذاكرة .

4: بالنسبة للهدف(OP1) يمكن ان يكون سجل أو موقع فى الذاكرة.

5: بالنسبة للمصدر(OP2) والهدف (OP1)لايمكن ان يكون كلاهما موقع بالذاكره.

مثال :

MOV AL , 8

MOV CL , 6

SUB AL , CL

الحل

AL= AL – CL

AL= 8 – 6

AL=02H

جدول متابعة:

AL	CL	الامر
؟؟	؟؟	
08H	؟؟	MOVAL , 8
08H	06H	MOV CL , 6
02H	06H	SUB AL , CL

3)الجمع مع الراية أو الحمل ADC (ADD WITH CARRY) CF

ويسمى ايضاً الجمع بالحمل.

الصيغة العامة لهذا ألامر هى ADC Destination,Source

ADC OP1 ,OP2

OP1=OP1+OP2+CF*

نفرض إن قيمة :-

OP1 تمثل الهدف (Destination)

OP2 تمثل المصدر (Source)

ملاحظات مهمه:

1: المقصود في CF* أي قيمة الراية CF التي كانت قبل عملية الجمع وليست القيمة الجديدة التي عملتها عملية الجمع الجديدة.

2: فى هذا الامر يتم جمع قيمة المصدر(OP2) وقيمة الهدف (OP1)وقيمة سجل الراية (CF) وتحفظ النتيجة في الهدف(OP1).

3 بالنسبة للمصدر (OP2) والهدف(OP1) يجب ان يكون كلاهما من نفس الحجم (8 bit – 16bit).

4:بالنسبة للمصدر(OP2) يمكن ان يكون قيمة فورية أو سجل أو موقع فى الذاكرة .

5: بالنسبة للهدف(OP1) يمكن ان يكون سجل أو موقع فى الذاكرة.

6: بالنسبة للمصدر(OP2) والهدف (OP1)لايمكن ان يكون كلاهما موقع بالذاكره.

مثال:

MOV AL,90H

ADD AL,85H

ADC AL,3

جدول متابعة:

ملاحظات	CF	AL	الامر
	?	??	
MOV لا تؤثر على الرايات	?	90H	MOV AL,90H
AL=90+85	1	15H	ADD AL,85H
AL=15H+3+1	0	19H	ADC AL,3

ملاحظة: ان النتيجة غير منطقية لانها بحاجة الى اكثر من ثمانية بتات

4)الطرح مع راية SBB (SUBTRUCT WITH CARRY) CF

ويسمى ايضاً الطرح بالإستلاف.

الصيغة العامة لهذا ألامر هى SBB Destination,Source

SBB OP1,OP2

$OP1=OP1-OP2-CF^{*}$

نفرض إن قيمة :-

OP1 تمثل الهدف (Destination)

OP2 تمثل المصدر (Source)

ملاحظات مهمه:

1: المقصود في CF* أي قيمة الراية CF التي كانت قبل عملية الطرح وليست القيمة الجديدة التي عملتها عملية الطرح الجديدة.

2: فى هذا الامر يتم طرح قيمة المصدر(OP2) وقيمة الهدف (OP1)وقيمة سجل الرية (CF) وتحفظ النتيجة في الهدف(OP1).

3 بالنسبة للمصدر (OP2) والهدف(OP1) يجب ان يكون كلاهما من نفس الحجم (8 bit – 16bit).

4:بالنسبة للمصدر(OP2) يمكن ان يكون قيمة فورية أو سجل أو موقع فى الذاكرة .

5: بالنسبة للهدف(OP1) يمكن ان يكون سجل أو موقع فى الذاكرة.

6: بالنسبة للمصدر(OP2) والهدف (OP1)لايمكن ان يكون كلاهما موقع بالذاكره.

مثال:

MOV AL,90H

SUB AL,85H

SBB AL,3

جدول متابعة:

الامر	AL	CF	ملاحظات
	??	?	
MOV AL,90H	90H	?	MOV لا تؤثر على الرايات
ADD AL,85H	05H	0	AL=90-85
ADC AL,3	02H	0	AL=05H-3-0

اوامر القسمة والضرب

اوامر الضرب:-

في لغة الاسمبلي توجد اوامر لضرب الاعداد الموجبة فقط واوامر لضرب الاعداد الموجهة أي(الموجبة والسالبة أي دون التقيد بالاشارة)

ضرب الاعداد الموجبة:-

نستعمل هنا امر الضرب MUL ولكن هنالك شروط يجب ان تتحقق عندما نريد ان نستعمل هذا الامر . MUL OP1

ملاحظات مهمة:

OP1:1 يمكن ان يكون قيمة فورية أو سجل أو موقع فى الذاكرة.

:2 يمكن ان يكون OP1 بطول(8 bit – 16bit).

:3 اذا كان OP1 من نوع بايت عندها تتم العملية التالية.

AX=AL*OP1

أي ان المعالج سوف يقوم بضرب OP1 فى المسجل AL وهذا المسجل يضع النتيجة في المسجل AX. لهذا السبب يجب ان نضع احدى القيم في المسجل AL وان نحفظ قيمة المسجل AH في مكان اخر قبل عملية الضرب لانه بعد عملية الضرب سوف يقوم المعالج بحذف القيمة القديمة ويضع مكانها القيمة الجديدة

مثال:

اكتب مقطع برنامج يضرب قيمة الخلية 20H في القيمة 8 ويحفظ النتيجة في المسجل CX

الحل:

```
MOV SI,20H

MOV AL,[SI]

MOV BL,8

MUL  BL

MOV CX,AX
```

اذا كان OP1 من نوع كلمة أي 16 بت عندها تتم العملية التالية

DX:AX=AX*OP1

عندها يقوم المعالج بضرب OP1 بقيمة المسجل AX ويحفظ النتيجة في المسجلين DX و AX لهذا السبب يجب ان نضع احدى القيم في المسجل AX وان نحفظ القيمة الموجودة في المسجل DX في مكان اخر قبل عملية الضرب.

مثال :

عليك ضرب العدد FFFFH بالعدد 4

الحل:

```
MOV AX,0FFFFH
MOV CX,4
MUL CX
```

ملاحظة: لقد اضفنا صفر امام العدد الذي وضعناه في المسجل AX لان القانون يقول كل عدد يبدأ بحرف من الحروف(A ,B ,C, D ,E , F) يجب اضافة صفر عن يساره.

ضرب الاعداد الموجهة

نفس الشروط التي ذكرتها سابقا ولكن بدلا من الامر MUL نستخدم IMUL

مثال :

اكتب مقطع برنامج ينفذ التمرين التالي

CX= -10 * DL

الحل:

```
MOV AL,-10
IMUL DL
MOV CX,AX
```

القسمة

تنقسم عمليات القسمة الى قسمين

1: قسمة الاعداد الموجبة فقط

2: قسمة الاعداد الموجهة(سالبة وموجبة)

قسمة الاعداد الموجبة

نستخدم في هذه الحالة الامر DIV

و هو بالشكل التالي:

DIV OP1

اذا كان OP1 من نوع بايت عندها تتم العملية التالية:

AX/OP1

يقوم المعالج بتقسيم القيمة الموجودة في المسجل AX (فقط هذا المسجل)

على OP1 ويتم حفظ الناتج في المسجلات التالية:

AL: يحفظ الناتج الصحيح

AH:يحفظ الباقي الصحيح

مثال:

قسم القيمة 9 على 4

الحل:

```
MOV AX,9
MOV BL,4
DIV BL
```

بعد تنفيذ هذا المقطع

AL=2 AH=1

ملاحظة: لا يجوز الاختزال في الباقي الصحيح حتى ولوأمكن ذلك

مثال:

```
MOV AX,8
MOV CL,6
DIV CL
```

AL=1 AH=2

صحيح انه 2/6=1/3 الا انه لا يجوز التبسيط

اذا كان OP1 من نوع 16 بت تتم العملية التالية

DX:AX/OP1

أي ان المعالج يتعامل مع المسجلين كانهما مسجلا واحدا ويقسمهما على OP1 ويحفظ النتيجة بالشكل التالي:

AX: يحتوي على الناتج الصحيح

DX:يحتوي على الباقي الصحيح

قسمة الاعداد الموجهة

هنا نستخدم نفس الشروط مثل الاعداد الموجبة ولكن الفرق الوحيد هو بدلا من الامر DIV نستخدم الامر IDIV

مثال:

احسب ما يلي

-120/12H

الحل:

```
MOV AX,-120
MOV BL,12H
IDIV BL
```

مثال:

اكتب مقطع برنامج يحسب

1234H/-1010

الحل:

```
MOV DX,0
MOV AX,1234H
MOV CX,-1010
IDIV CX
```

ملاحظة:

لقد وضعنا القيمة صفر في المسجل DX قبل عملية القسمة لان المعالج سوف يستخدم القيمة الموجودة فيه عند القسمة ويمكن ان تكون قيمة مختلفة عن الصفر وحتى نتأكد ان القيمة صفر وضعنا الصفر في DX

ثانياً العمليات المنطقية

النتيجة من هذه العمليات تكون اما صدق TRUE والتي نعبر عنها في القيمة 1 . او تكون النتيجة من نوع كذب FALSE والتي نعبر عنها بالقيمة 0.

1) الامر OR

هنا يكفي شرطا واحدا ان يتحقق فنحصل على نتيجة صدق.

جدول الامر OR

النتيجة A+B	B	A
0	0	0
1	1	0
1	0	1
1	1	1

ملاحظات :-

1: A,B عبارة عن بتات .

2: يأخذ هذا الامر المصدر والهدف OR OP1 , OP2

3: يجب تحويل قيم المصدر والهدف الى الميزان الثنائي حتى نتمكن من التعامل مع القيم.

مثال:

أوجد قيمة كل مسجل من المسجلات بعد تنفيذ العملية المنطقية التالية:

OR AL , BL

حيث ان:

AL=2FH

BL=52H

الحل:

علينا ان نحول الهدف والمصدر الى الميزان الثنائي

AL=2FH= 00101111

BL=52H= 01010010

العدد الاول	0	0	1	0	1	1	1	1
العدد الثاني	0	1	0	1	0	0	1	0
النتيجة =	0	1	1	1	1	1	1	1

بماً ان المسجل BL هو المصدر فان قيمته لا تتغير وتبقى BL=52H

والمسجل AL هو الهدف اذاً التأثير يتم عليه حسب النتيجة AL=7FH

2) الامر AND

هنا يكفي ان لا يتحقق شرطا واحدا حتى نحصل على نتيجة كذب

جدول الامر AND

النتيجة A*B	B	A
0	0	0
0	1	0
0	0	1
1	1	1

ملاحظات :-

1: A,B عبارة عن بتات .

2: يأخذ هذا الامر المصدر والهدف AND OP1 , OP2

3: يجب تحويل قيم المصدر والهدف الى الميزان الثنائي حتى نتمكن من التعامل مع القيم.

مثال:

AL=27H

BL=56H

أوجد قيمة كل مسجل من المسجلات بعد تنفيذ العملية المنطقية التالية:

AND AL , BL

حيث ان :

AL=27H

BL=56H

الحل:

علينا ان نحول الهدف والمصدر الى الميزان الثنائي

AL=27H= 00100111

BL=56H= 01010110

العدد الاول	0	0	1	0	0	1	1	1
العدد الثاني	0	1	0	1	0	1	1	0
=النتيجة	0	0	0	0	0	1	1	0

بما ان المسجل BL هو المصدر فان قيمته لا تتغير وتبقى BL=56H

والمسجل AL هو الهدف اذاً التأثير يتم عليه حسب النتيجة AL=06H

3) الامر TEST

هنا يكفي ان لا يتحقق شرطا واحدا حتى نحصل على نتيجة كذب .

جدول الامر TEST

A	B	النتيجةA*B
0	0	0
0	1	0
1	0	0
1	1	1

ملاحظات :-

1:A,Bعبارة عن بتات .

2: يأخذ هذا الامر المصدر والهدف TEST OP1 , OP2

3: يجب تحويل قيم المصدر والهدف الى الميزان الثنائي حتى نتمكن من التعامل مع القيم.

4: يعمل هذا الامر مثل الامر AND ولكن مع فرق بسيط وهو انه لا يؤثر على الهدف ولا على المصدر وانما يؤثر فقط على الرايات ويقوم بتغيير بعض الرايات

مثال:

أوجد قيمة كل مسجل من المسجلات بعد تنفيذ العملية المنطقية التالية:

TEST AL , BL

حيث ان :

AL=27H

BL=56H

الحل:

علينا ان نحول الهدف والمصدر الى الميزان الثنائي

AL=27H= 00100111

BL=56H= 01010110

العدد الاول	0	0	1	0	0	1	1	1
العدد الثاني	0	1	0	1	0	1	1	0
=النتيجة	0	0	0	0	0	1	1	0

بماً ان المسجل BL هو المصدر فان قيمته لا تتغير وتبقى BL=56H

والمسجل AL هو الهدف ولكن الامر TEST لا يؤثر على الهدف المسجل يحافظ على قيمته السابقة AL=27H

اذاً ما هي الفائدة منه؟

الجواب : في بعض الاحيان نحتاج الى فحص بت معينة دون ان نؤثر على القيمة الاصلية. فنستخدم هذا الامر

4)الامر XOR

هنا اذا كانت البتات مختلفة النتيجة 1 واذا كانت متشابهة النتيجة 0

جدول الامر XOR

النتيجة	B	A
0	0	0
1	1	0
1	0	1
0	1	1

ملاحظات :-

1:A,Bعبارة عن بتات .

2: يأخذ هذا الامر المصدر والهدف TEST OP1 , OP2

3: يجب تحويل قيم المصدر والهدف الى الميزان الثنائي حتى نتمكن من التعامل مع القيم.

مثال:

AL=A7H

BL=76H

جد قيمة كل مسجل من المسجلات بعد تنفيذ العملية المنطقية التالية:

XOR AL , BL

حيث ان :

AL=A7H

BL=76H

الحل:

علينا ان نحول الهدف والمصدر الى الميزان الثنائي

AL=A7H= 10100111

BL=76H= 01110110

العدد الاول	1	0	1	0	0	1	1	1
العدد الثاني	0	1	1	1	0	1	1	0
النتيجة =	1	1	0	1	0	0	0	1

بماً ان المسجل BL هو المصدر فان قيمته لا تتغير وتبقى BL=76H

والمسجل AL هو الهدف اذاً التأثير يتم عليه حسب النتيجة AL=D1H

5)الامر NOT

هنا يتم عكس البتات البت التي قيمتها 1 تأخذ القيمة 0 والبت التي قيمتها 0 تأخذ القيمة 1

جدول الامر not

النتيجة	A
1	0
0	1

ملاحظات :-

1:A عبارة عن بت.

2 يأخذ هذا الامر OPERAND واحد فقط NOT OP1

3 يجب تحويل قيمته الى الميزان الثنائي حتى نتمكن من التعامل معه.

مثال:

أوجد قيمة كل مسجل من بعد تنفيذ العملية المنطقية التالية:

NOT AL

حيث ان:

AL=A7H

الحل:

علينا ان نحول الى الميزان الثنائي

AL=A7H= 10100111

العدد الاول	1	0	1	0	0	1	1	1
النتيجة	0	1	0	1	1	0	0	0

المسجل AL هو الذي يتأثر حسب النتيجة AL=58H

اللهم علمنا ما ينفعنا وانفعنا بما علمتنا

وانفع الناس بنا واغفر لنا وارحمنا إنك أنت الغفور الرحيم

م / محمد عيـد الفائـدى

Alfayde 2002@yahoo.com